

1 SYSTEM AND METHOD FOR ACCESSING READERS AND OTHER I/O
2 DEVICES BY PROGRAMS

3 FIELD OF INVENTION

4 The present invention relates program specific configuration
5 of several physical or logical attachments in a system. It
6 more particularly relates to controlling access of programs
7 to various I/O devices considering use restrictions and
8 priority orders assigned to the programs.

9 BACKGROUND OF THE INVENTION

To write and to read data to a smart card or to execute a command on a smart card, it is necessary to use a connection with the card. The connection with the smart card is made by using a reader. Readers of the same manufacturer use some software support (driver) for communication with programs by using a standard reader API (Programming Interface) preferably. In this case user can replace one reader with another compatible reader without changing code of the program.

19 However, many of the programming interfaces of the readers
20 (driver) currently available are not fully standardized.
21 Thus readers of different or the same manufacturers may be
22 present concurrently at the same system for different

1 programs. Furthermore, there are use restrictions of certain
2 types of smart cards per reader or for security relevant
3 programs in which the use of more than one card in one
4 session is prohibited.

5 If several readers are installed on one system, the user is
6 not able to select the right reader when a program requests
7 to insert a smart card. Furthermore, if additional readers
8 are installed after set up of the programs, the access
9 priority of the installed readers may be changed. This may
10 cause errors of the programs or could confuse the user when
11 asked to insert a smart card into a reader. If several
12 readers with different drivers share a logical I/O port,
13 errors may occur depending on which reader is actually
14 connected and powered on. Readers reserved for security
15 relevant programs (e.g. used for user identification and
16 authentication at system start) may be accessible for other
17 programs which may cause the disruption of system services
18 when a reader is locked by another reader or if the smart
19 card having sensitive information is removed. This restricts
20 the use of systems in which several parallel running
21 programs having access to different readers having standard
22 or nonstandard reader API.

23 US Patent No. 3810105 discloses a computer input-output
24 system in which peripheral devices (e.g. readers) cooperate
25 with hardware input-output processors independent from the
26 central processor of the computer for handling the transfer
27 of data between peripheral devices. Signal communication

1 runs through special transmission facilities which include
2 separate communication paths for the input-output
3 processors, separate communication paths for control and
4 data signals, and separate communication paths for
5 determining priority of operations among several
6 input-output processors and the CPU at memory. The devices
7 are controlled by device controller including subcontrollers
8 which together with a portion of the input-output processors
9 provides a communication interface configuration between
10 devices and input-output processors. There is no teaching or
11 suggestion in that patent how readers may be accessed by
12 user defined access conditions.

13 **SUMMARY OF THE INVENTION**

14 It is therefore an aspect of the present invention to
15 provide an improved access control mechanism to readers or
16 other I/O devices by programs installed on one system having
17 access to various readers or other I/O devices avoiding the
18 disadvantages of the other access control mechanisms.

19 The present invention allows a program specific
20 configuration of several physical or logical readers or
21 other I/O devices (hereinafter called readers) by using a
22 configuration tool and a reader access layer. The
23 configuration tool allows to specify access rights and
24 priority rights for each single reader in conjunction with
25 each single program. A program may or may not be granted

1 access rights to a reader or a program can be granted access
2 to several readers using a priority number for selecting the
3 reader to be accessed at first.

4 The present invention secures that previously defined access
5 rights and priorities between readers and programs defined
6 in the reader access list remain unchanged independently
7 when new readers are added. Amendments are allowed by the
8 configuration tool only.

9 The present invention is especially valuable in the future
10 when more and more security programs are using a mix of
11 readers attached via the standard serial ports and an
12 universal serial bus (USB) with automatic configuration at
13 plug-in called „plug and play“. Attaching a new reader will
14 then not disrupt existing relationships between readers and
15 programs.

16 **BRIEF DESCRIPTION OF THE DRAWINGS**

17 These and other aspects, features, and advantages of the
18 present invention will become apparent upon further
19 consideration of the following detailed description of the
20 invention when read in conjunction with the drawing figures,
21 in which:

22 Fig. 1(A) shows a prior art reader access list illustrating
23 prior art access control mechanism;

1 ;
2 Fig. 1(B) shows a reader access list according to Fig. 1(A)
3 with the difference that two smart cards are
4 inserted;
5 Fig. 1(C) shows a reader access list according to Fig. 1(B)
6 with the difference that a new reader has been
7 installed;
8 Fig. 2 shows an example of basic components of the
9 present invention implemented in a typical
10 communication architecture;
11 Fig. 3 shows an example flow chart which describes the
12 inventive access control mechanism;
13 Figs. 4(A-H) show examples of reader access lists
14 illustrating the present invention; and
15 Fig. 5 shows an example of a reader list display which
16 may be used by the present invention.

17 DESCRIPTION OF THE INVENTION

18 The present invention allows a program specific
19 configuration of several physical or logical readers or
20 other I/O devices (hereinafter called readers) by using a
21 configuration tool and a reader access layer. The

1 configuration tool allows to specify access rights and
2 priority rights for each single reader in conjunction with
3 each single program. A program may or may not be granted
4 access rights to a reader or a program can be granted access
5 to several readers using a priority number for selecting the
6 reader to be accessed at first. In a case of failing of a
7 reader (e.g. failing serial connection, failing battery,
8 missing smart card) to be accessed at first, the reader with
9 the next highest priority number has to be selected as
10 backup-reader. Programs having no assigned priority using
11 the standard priority specified in the reader access list.
12 The reader access layer communicates with each program
13 directly, e.g. receives all requests from program seeking
14 access to a readers, calls up the reader access list for the
15 requesting program, checks the access rights and the
16 priority order for the available readers (e.g. which reader
17 has to be accessed at first if more than one readers are
18 accessible) and returns a response to the requesting program
19 containing information for accessing the active reader with
20 the highest priority.

21 The present invention secures that previously defined access
22 rights and priorities between readers and programs defined
23 in the reader access list remain unchanged independently
24 when new readers are added. Amendments are allowed by the
25 configuration tool only.

26 The present invention is especially valuable in the future
27 when more and more security programs are using a mix of

1 readers attached via the standard serial ports and an
2 universal serial bus (USB) with automatic configuration at
3 plug-in called „plug and play“. Attaching a new reader will
4 then not disrupt existing relationships between readers and
5 programs.

6 Figures 1(A-C) show examples of reader access lists for
7 smart card readers in a system using a PKSC#11 program
8 interface as used by prior art implementations. There are
9 three different classes of readers:

10 1. Direct controlled readers with vendor specific
11 device driver(s)
12 2. PC/SC registered readers with standardized device
13 driver(s)
14 3. Virtual (software emulated) readers with „virtual
15 smart cards“.

16 Each of these reader classes have their own default access
17 priority scheme:

18 1. The direct controlled readers are defined in a
19 special file named e.g. „Readers.cfg“
20 2. The PC/SC readers are prioritized in alphabetic
21 order of the manufacturer name appended by a serial
22 number assigned by the operation system at reader
23 installation time

1 3. The virtual readers are sorted in alphabetic order
2 of the names assigned at virtual smart card creation
3 time.

4 All programs using the same API have access to all readers
5 presented in e.g. in the PKCS#11 API in a slot list. The
6 program can check if a smart card is inserted in a reader or
7 if a virtual smart cards (VSC) is enabled or disabled.

8 In Figures 1(A-C) two programs using the readers which are
9 listed above. All programs (AA, BB) have the same access
10 rights as shown in columns, Appl. AA and Appl. BB' in Figure
11 1(A). If a card is inserted as shown in Figure 1(B), the
12 access priority (column order) is changed so that the first
13 card detected is now in reader „A- Vendor-Terminal“ instead
14 of „X-Vendor Terminal“.

15 In Figure 1(C) a new PC/SC reader of the same „A-vendor“ is
16 added with a smart card inserted. This reader will be
17 placed by the PC/SC operating system in a table of available
18 reader directly behind the other reader from the same
19 manufacturer and same reader type with suffix „2“. This will
20 change the reader access priority (Column order) of all
21 following readers for each program. In summary, these
22 examples show that the program cannot be sure which reader
23 and/or smart card is selected in priority when readers are
24 replaced, added or removed.

1 Figure 2 shows the basic components of the present invention
2 namely configuration tool and reader access layer. The
3 main function of the configuration tool (4) is to specify a
4 reader access list used by the reader access layer (6). In a
5 advantageous embodiment, each program will have its own
6 reader access list (8,10,12). Another implementation may be
7 that all programs are listed in a common reader access list.
8 The reader access list (8,10,12) is advantageously laid down
9 in a file and permanently stored in a nonvolatile storage
10 media of the system and may be called up by the
11 configuration tool (4) or by the reader access layer (6) by
12 its file name. Amendments in the reader access list
13 (8,10,12) are allowed by the configuration tool (4) only.
14 New installed readers will not automatically change the
15 access rights or priority order of the available readers
16 (16) without using the configuration tool (4). The reader
17 access list (8,10,12) contains configuration data relating
18 access rights and priority rights for each single reader
19 (14,16) in conjunction with each single program (2). Thus, a
20 program (2) may or may not be granted access rights to a
21 reader (14, 16) or a program (2) can be granted access to
22 several readers using a priority identifier for selecting
23 the reader (14,16) to be accessed at first. In a case of
24 failing of a reader (e.g. failing serial connection, failing
25 battery, missing smart card) to be accessed at first the
26 reader with the next highest priority number has to be
27 selected as backup-reader.

1 The reader access layer (6) communicates with each program
2 directly, e.g. receives all requests from programs (2)
3 seeking access to a reader, calls up the reader access list
4 (8,10,12) for the requesting program (2), checks the access
5 rights and the priority order for the available readers
6 (e.g. which reader has to be accessed at first if more than
7 one readers are accessible) and returns a response to the
8 requesting program (2) containing information for accessing
9 the active reader (14,16) with the highest priority.

10 A standard implementation of the present is that the both
11 basic components are installed on one system. However it may
12 be possible that the present invention may be used in a
13 client- server architecture by distributing both components
14 in a client and a server system. For example, the
15 configuration tool (4) and the reader access layer (6) could
16 be installed on the server side and the programs (2) could
17 be installed on the client side. Another implementation may
18 be that after each new configuration of the reader access
19 list (8,10,12) on the server side, the updated reader access
20 list will be send to the client. This implementation however
21 requires that the reader access layer (6) is available on
22 the server as well on the client system.

23 Figure 2 shows the basic components of inventive access
24 control mechanism in a system environment comprising for
25 example three application programs AA, BB, CC (2) and four
26 physical (16) and two logical readers (14). The logical

1 readers (14) are two virtual readers (14) with virtual smart
2 cards.

3 The virtual reader including the virtual smart cards may be
4 created by the configuration tool. The virtual smart cards
5 may be either enabled or disabled emulating the „Inserted /
6 removed“ status of a real smart card. For each registered
7 application program (AA, BB, CC) for which a reader access
8 list exists the configuration tool allows to specify access
9 rights and priority rights for a specific application
10 program (AA, BB, CC).

11 In the reader access list (8) for Appl.AA the priority (1)
12 has been assigned to the ,PC/SC Reader A'(16) and the
13 priority (2) to the virtual smart card 1(14). All other
14 readers are not accessible for Appl.AA, which means they
15 have the priority (0). If either the reader PC/SC Reader A
16 (16) is not available or one of the portable hardware token
17 ,Token 1' or , Token 2' are not inserted in this Reader
18 (16), the reader in the list with the next lower priority is
19 used which is in this example the „virtual smart card“ (14).

20 For the Appl. BB the reader ,CT-API Reader Y'(16) has the
21 priority (1), ,Virtual Smart Card 2' (16) the priority (2),
22 and ,PC/SC Reader B'(16) priority (3). For this program only
23 these readers are accessible. In cases wherein only one
24 token is used by the program, the ,PC/SC Reader B' with the
25 lowest priority is only used if the ,CT-API Reader Y' is not
26 available and the ,Virtual Smart Card' is disabled. For all

1 other programs accessing readers the „standard priority
2 list“ is used.

3 Specifying a (0) in this list means that this reader is
4 available only for registered programs with the reader
5 selected. In Figure 2 these are ‚PC/SC Reader A‘ and
6 ‚Virtual Smart Card 1‘. The remaining readers are assigned
7 in the sequence of their assigned priorities 1 to n.

8 Figure 3 shows a flow chart which describes the inventive
9 access control mechanism as used by the present invention.

10 1. Program sends a request to the access layer for
11 accessing a reader wherein the access layer examines
12 whether the program is already registered.

13 2. Reader access layer examines whether a reader
14 access list is available for the requesting program
15 (2). If there is no reader access list available (the
16 requesting program is not registered) the access layer
17 calls up a standard reader access list used for
18 unregistered programs only (4).

19 3. If the requesting program is registered, the
20 reader access layer calls up the reader access list (6)
21 and carries out a routine according to step 4) by
22 ignoring not assigned readers (8). This applies
23 accordingly for the standard priority.

DE920000075US1

1 4. The routine will start with the reader with the
2 highest priority (8). If that reader is not available
3 (e.g. smart card is not inserted or the reader is out
4 of order), the reader with the next priority will be
5 selected and so on until an active reader with a smart
6 card inserted has been identified (10). In that case
7 the program receives a return with a pointer to the
8 active reader (12). If no reader is available, the
9 program receives the information that „no active reader
10 has been found“ (14). The routine for determining the
11 active reader with the highest priority will be applied
12 for the standard priority accordingly (7).

13 The above access control mechanism is applied by the reader
14 access layer only. A further embodiment could be that the
15 routine according to step 4) is carried out by the program
16 itself. In that case the reader access layer should provide
17 information of the assigned readers with their priority
18 order to the program.

19 Figures 4 (A-H) show examples of reader access lists for
20 illustrating the present invention.

21 In Figure 4 (A) reader access list is shown containing
22 readers arranged by the operating system in a priority order
23 (see left column). This priority order is generated by the
24 operating system without using the idea of the present
25 invention.

1 In Figure 4(B) a reader access list according to the present
2 invention is shown containing all available readers with the
3 three priority columns. The administrator may specify the
4 desired priority for general programs not registered
5 (standard priority column) and for each program (Appl.AA,
6 Appl.BB priority column) by entering a digit 1 to n.
7 Entering the digit 0 will disable the reader for that
8 program. When the reader access list has been completed for
9 each program, as shown in FIG. 4(B), the priority sequence
10 are different for all three readers groups. Some readers are
11 not accessible for either Appl.AA or Appl.BB (indicated by a
12 ,0'). Optionally, the reader provided reader names may be
13 customized for each program.

14 In Figure 4(C) the readers are sorted by the specified
15 "standard priority" and the program view and priority
16 sequence of the different readers are shown for Appl.AA and
17 Appl.BB assuming the status of the readers as shown in
18 Figure 4(A). Inserting a smart card into ,A-Vendor Terminal
19 1' will now effect only the program Appl.BB as shown in
20 Figure 4(D), the access priority per program is based on the
21 specification in the reader access list.

22 In the next example a new PC/SC terminal (A-Vendor Terminal
23 2) is added to the system and will be automatically
24 configured by the operating system. Without the present
25 invention it would have the priority 4 as shown in Figure
26 4(E). In the inventive reader access list the ,A-Vendor
27 Terminal 2' is listed with the initial standard priority

1 using the last previously specified priority number plus one
2 (priority is 7) as shown in Figure 4(F). For the programs
3 Appl.AA and BB that reader is not available indicated by
4 number ,0'.

5 As shown in Figure 4(G) the addition of this new reader will
6 have no effect on the operation of the Appl.AA/BB. For
7 programs using the standard priority it will appear as last
8 reader in the priority order. If this reader should be made
9 available to the Appl.AA/BB, the administrator has to change
10 the priority number from ,0' to any other priority number.

11 If this reader should be used by a new program a new
12 column, e.g. Appl.CC priority, has to be added to the reader
13 access list as shown in Figure 4H. In addition, the standard
14 priority may be changed by giving a certain priority for the
15 available readers.

16 Figure 5 shows an example of a reader list display
17 advantageously used by the present invention. The
18 configuration utility displays all attached real and virtual
19 readers for the system administrator. In the present example
20 two physical readers (Gemplus GPR 400 0; TOWITKOKO
21 CHIPDRIVE) and one virtual reader (IBM Virtual Smartcard)
22 are installed. In the TOWITKOKO CHIPDRIVE a smartcard is
23 already inserted. This reader is now in an active status.
24 This is indicated by specific insertion symbol. The
25 remaining readers are in not active status. Out of this list

1 the administrator has to define the priority order in which
2 the readers are presented to the programs.

3 Although, the present invention has been described primarily
4 with respect to readers only, every suitable I/O device with
5 the functionality to communicate to different applications
6 and which may be selected by user defined access conditions
7 are also applicable for accomplishing the present invention.
8 For example the I/O device may be a communication link,
9 cryptographic adapter, printer, etc. Thus the word reader
10 as used herein includes any I/O device.

11 The present invention can be realized in hardware, software,
12 or a combination of hardware and software. A visualization
13 tool according to the present invention can be realized in a
14 centralized fashion in one computer system, or in a
15 distributed fashion where different elements are spread
16 across several interconnected computer systems. Any kind of
17 computer system - or other apparatus adapted for carrying
18 out the methods and/or functions described herein - is
19 suitable. A typical combination of hardware and software
20 could be a general purpose computer system with a computer
21 program that, when being loaded and executed, controls the
22 computer system such that it carries out the methods
23 described herein. The present invention can also be
24 embedded in a computer program product, which comprises all
25 the features enabling the implementation of the methods
26 described herein, and which - when loaded in a computer
27 system - is able to carry out these methods.

1 Computer program means or computer program in the present
2 context include any expression, in any language, code or
3 notation, of a set of instructions intended to cause a
4 system having an information processing capability to
5 perform a particular function either directly or after
6 conversion to another language, code or notation, and/or
7 reproduction in a different material form.

8 Thus the invention includes an article of manufacture which
9 comprises a computer usable medium having computer readable
10 program code means embodied therein for causing a function
11 described above. The computer readable program code means
12 in the article of manufacture comprises computer readable
13 program code means for causing a computer to effect the
14 steps of a method of this invention. Similarly, the present
15 invention may be implemented as a computer program product
16 comprising a computer usable medium having computer readable
17 program code means embodied therein for causing a function
18 described above. The computer readable program code means
19 in the computer program product comprising computer readable
20 program code means for causing a computer to effect one or
21 more functions of this invention. Furthermore, the present
22 invention may be implemented as a program storage device
23 readable by machine, tangibly embodying a program of
24 instructions executable by the machine to perform method
25 steps for causing one or more functions of this invention.

1 It is noted that the foregoing has outlined some of the more
2 pertinent objects and embodiments of the present invention.
3 This invention may be used for many applications. Thus,
4 although the description is made for particular arrangements
5 and methods, the intent and concept of the invention is
6 suitable and applicable to other arrangements and
7 applications. It will be clear to those skilled in the art
8 that modifications to the disclosed embodiments can be
9 effected without departing from the spirit and scope of the
10 invention. The described embodiments ought to be construed
11 to be merely illustrative of some of the more prominent
12 features and applications of the invention. Other
13 beneficial results can be realized by applying the disclosed
14 invention in a different manner or modifying the invention
15 in ways known to those familiar with the art.

100-4525-1025-04